

## **REMARKS**

Applicant is in receipt of the Office Action mailed July 5, 2007. Claims 2-5 have been cancelled. Claims 1, 6, 7, 8, and 17-20 have been amended. Thus, claims 1 and 6-23 are pending in the case. Reconsideration of the present case is earnestly requested in light of the following remarks.

### **Telephone Interview Summary**

On Friday, August 24, 2007, a telephone interview was conducted between the Examiner, Jeffrey Hood (#35,198), and Mark Williams (#50,658). The claimed invention scope and cited art was discussed. Applicant proposed amending the independent claims to clarify that the GUI element/tooltip is displayed in response to the user hovering a mouse cursor over the expression in the source code. The Examiner stated that an updated search would be required. Applicant agreed to submit the amendment with a Request for Continued Examination.

### **Section 103 Rejections**

Claims 1 and 18-21 were rejected under 35 U.S.C. 103(a) as being unpatentable over PerfectXML (<http://www.perfectxml.com/nr/aspnetdebug.pdf>) in view of Michelman et al (US Pat. 6,907,580, “Michelman”).

Claim 1 recites, in part, that source code for the program is displayed on a display during execution of the program, that the GUI element is displayed proximate to the expression, that user input is received to the GUI element modifying the displayed value, that the expression in the program is set to the new value, and the program continues execution in accordance with the new value of the expression. These limitations are simply not taught or suggested in the cited art. In other words, the cited art makes no mention of modifying a value in an executing program and then continuing execution of the program in accordance with the new value.

As discussed in the previous Response, the Office Action asserts that Michelman is directed to a “graphical user interface (GUI) for debugging a program”. Applicant

respectfully disagrees, noting that Michelman nowhere mentions or even hints at debugging a program. In the Response to Arguments, the Examiner asserts that “Michelman mentions that the tooltip can be implemented in various ways”, citing col.7:25. However, Applicant notes that Michelman nowhere even mentions “tooltip” or “debugging” at all, nor displaying source code on a display. Nor does Michelman’s described user interface element have the same functionality as a tooltip. Applicant submits that Michelman nowhere discloses the particular features and limitations of Applicant’s claimed invention.

As Michelman’s Abstract makes clear, Michelman is directed to a moveable displayed user interface element that can be used to select an item displayed on a user interface, after which an area of the moveable displayed user interface element can be activated to perform an operation on the selected item or items. More particularly, Michelman’s UI elements are described as interfaces for performing specified operations on items upon activation of an area *on the user interface element*, e.g., by pressing a button, etc. Additionally, Michelman’s UI element is moved around on the screen to select the items to be operated on, e.g., by the user dragging the element around with a mouse/cursor. This is in direct contrast with a tooltip, which, as is well-known to those of skill in the art of GUIs, is a GUI element that is displayed automatically and dynamically, e.g., when the user hovers a cursor over an item, and is *not* dragged around on the screen via a pointing device. Applicant has amended the independent claims as indicated above to further emphasize this aspect of Applicant’s claimed invention.

Amended claim 1 recites:

1. A memory medium which stores program instructions implementing a graphical user interface (GUI) for debugging a program, wherein, during execution of the program, the program instructions are executable by a processor to perform:

displaying source code for the program on a display;

receiving first user input hovering a mouse cursor over an expression in the source code;

in response to said hovering the mouse cursor over the expression, displaying a value of the expression in a GUI element proximate to the expression;

receiving second user input to the GUI element modifying the displayed value, thereby specifying a new value for the expression; and

setting the expression in the program to the new value, wherein the program continues execution in accordance with the new value of the expression.

Applicant respectfully submits that neither PerfectXML nor Michelman teaches or suggests **in response to said hovering the mouse cursor over the expression, displaying a value of the expression in a GUI element proximate to the expression; receiving second user input to the GUI element modifying the displayed value, thereby specifying a new value for the expression; and setting the expression in the program to the new value, wherein the program continues execution in accordance with the new value of the expression**, as recited in amended claim 1.

Rather, PerfectXML discloses a standard tooltip for displaying a value, and a watch window for monitoring and possibly changing values of variables. Nowhere does PerfectXML disclose these claimed features and limitations of claim 1.

Nor does Michelman disclose a GUI element with the above claimed functionality. For example, Applicant notes that Michelman is directed to a GUI element that the user drags over items, that then allows the user to activate the GUI element to perform some operation on a selected item. Thus, even when Michelman's GUI element is used to display a value, the value is not displayed in response to hovering a mouse cursor over the item, but rather, the value is displayed in response to user activation of the GUI element, e.g., via pressing a button on the GUI element. Moreover, neither Michelman nor PerfectXML discloses editing the value of an expression in source code for a program in the manner of claim 1, i.e., via user input to a GUI element displayed in response to hovering a mouse cursor over the expression. The cited references also do not disclose continuing execution of the program using the new value as recited in the present claims.

Thus, taken singly or in combination, PerfectXML and Michelman fail to disclose these features and limitations of claim 1.

Thus, for at least these reasons, Applicant submits that claim 1 and those claims dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claims 18, 19, 20, and 21 each includes similar limitations as claim one, and so the above arguments apply with equal force to these claims. For example, method claim 18 and system claim 19 each specifically refers to a “tooltip” with the functionality of the “GUI element” of claim 1. Memory medium claim 21 refers instead to a “window proximate to the expression, wherein the window is operable to display a value of the indicated expression, wherein the window does not include window title bars or menus”, which also includes the functionality of the GUI element of claim 1.

Thus, for at least the reasons provided above, Applicant submits that claims 18, 19, 20, and 21, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Removal of the section 103 rejection of claims 1 and 18-21 is respectfully requested.

Claims 1-23 were rejected under 35 U.S.C. 103(a) as being unpatentable over PerfectXML (<http://www.perfectxml.com/nr/aspnetdebug.pdf>) in view of East Tennessee State University ([http://csciwww.etsu.edu/blair/Using\\_Debugger.htm](http://csciwww.etsu.edu/blair/Using_Debugger.htm), “ETSU”). Applicant respectfully disagrees.

Applicant respectfully submits that neither PerfectXML nor ETSU teaches or suggests **in response to said hovering the mouse cursor over the expression, displaying a value of the expression in a GUI element proximate to the expression; receiving second user input to the GUI element modifying the displayed value, thereby specifying a new value for the expression; and setting the expression in the program to the new value, wherein the program continues execution in accordance with the new value of the expression**

As Applicant has discussed previously, the cited p.108, paragraph 3, of PerfectXML describes displaying and editing a value of an expression in a program via a watch window, but nowhere discloses doing so via a GUI element displayed proximate to

the expression in response to hovering a cursor over the expression. Similarly, as also discussed previously, while ETSU does describe displaying a variable value (in a tooltip) in response to holding the mouse cursor over the variable name (referred to as “Mouse Over”), nowhere does the cited art disclose receiving user input to the tooltip modifying the displayed value, thereby specifying a new value. In fact, nowhere do PerfectXML or ETSU disclose receiving user input to a tooltip or any other GUI element displayed in response to hovering a cursor over the expression at all. Nor do PerfectXML or ETSU disclose setting the expression in the program to the new value in response to such input to the GUI element, where the program continues execution in accordance with the new value of the expression.

In Response to Arguments, the Examiner asserts that original claim 1 was broad enough to cover ETSU’s use of a watch window to modify the value displayed in a tooltip. Applicant has amended the independent claims to specify that the GUI element (or tooltip or window, as claimed) is displayed proximate to the expression in response to hovering a mouse cursor over the expression, and that the second user input modifying the value is received to this GUI element, and respectfully submits that neither PerfectXML nor ETSU teaches these features.

Thus, for at least the reasons provided above, Applicant submits that PerfectXML and ETSU, taken singly or in combination, fail to teach or suggest all the features and limitations of claim 1, and so claim 1 and those claims dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claims 18, 19, 20, and 21 each includes similar limitations as claim one, and so the above arguments apply with equal force to these claims. For example, method claim 18 and system claim 19 each specifically refers to a “tooltip” with the functionality of the “GUI element” of claim 1. Memory medium claim 21 refers instead to a “window proximate to the expression, wherein the window is operable to display a value of the indicated expression, wherein the window does not include window title bars or menus”, which also includes the functionality of the GUI element of claim 1.

Thus, for at least the reasons provided above, Applicant submits that claims 18, 19, 20, and 21, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Removal of the section 103 rejection of claims 1 and 18-21 is respectfully requested.

Applicant also asserts that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the independent claims have been shown to be patentably distinct, a further discussion of the dependent claims is not necessary at this time.

## **CONCLUSION**

In light of the foregoing amendments and remarks, Applicant submits the application is now in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above-referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. The Commissioner is hereby authorized to charge any fees which may be required or credit any overpayment to Meyertons, Hood, Kivlin, Kowert & Goetzel P.C., Deposit Account No. 50-1505/5150-82801/JCH.

Also filed herewith are the following items:

- Request for Continued Examination
- Terminal Disclaimer
- Power of Attorney By Assignee and Revocation of Previous Powers
- Notice of Change of Address
- Other:

Respectfully submitted,

/Jeffrey C. Hood

Jeffrey C. Hood, Reg. #35198  
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC  
P.O. Box 398  
Austin, TX 78767-0398  
Phone: (512) 853-8800  
Date: 2007-08-29 JCH/MSW